Package ‘intRegGOF’

October 13, 2022

Version 0.85-5
Date 2018-05-21
Title Integrated Regression Goodness of Fit
Author Jorge Luis Ojeda Cabrera <jojeda@unizar.es>
Maintainer Jorge Luis Ojeda Cabrera <jojeda@unizar.es>
Description Performs Goodness of Fit for regression models using Integrated Regression method. Works for several different fitting techniques.
Depends R (>= 2.5.0), graphics, stats
License GPL (>= 2)
NeedsCompilation yes
Repository CRAN
Date/Publication 2018-05-22 12:28:58 UTC

R topics documented:

<table>
<thead>
<tr>
<th>Package</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>anovarIntReg</td>
<td>Integrated Regression Goodness of Fit</td>
<td>1</td>
</tr>
<tr>
<td>Core Functions</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>intRegGOF</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>plot</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>Utility Functions</td>
<td></td>
<td>8</td>
</tr>
</tbody>
</table>

Index

<table>
<thead>
<tr>
<th>Package</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>anovarIntReg</td>
<td>Integrated Regression Goodness of Fit</td>
</tr>
</tbody>
</table>

Description

Integrated Regression Goodness of Fit to test the adequacy of different model to represent the regression function for a given data.
Usage

```r
anovarIntReg(objH0, ..., covars = NULL, B = 499,
LINMOD = FALSE, INCREMENTAL = FALSE)
## S3 method for class 'anovarIntReg'
print(x, ...)
```

Arguments

- `objH0`:
  An object of class `lm`, `glm` or `nls` which will be considered as null hypotheses model or the base reference model when `INCREMENTAL` is set to TRUE.

- `...`:
  One or more objects of class `lm`, `glm` or `nls`.

- `covars`:
  Names of continuous (numerical) variates used to compute Integrated Regression. They should be variables contained in the data frame used to compute the regression fit. When NULL it is obtained as the max. number of different covariates in all tested models. It also can be a formula like `~x1+x2+...`

- `B`:
  Bootstrap resampling size.

- `LINMOD`:
  When TRUE and if `obj` is an object of class `print.intRegGOF`, Linear Model matrix fitting equations are used.

- `INCREMENTAL`:
  When FALSE all models in ... are tested against `objH0`, while when TRUE each of the models are checked against the next one starting in `objH0`.

- `x`:
  An object of class `anovarIntReg`.

Details

This function implements the test

\[ H_0 : m \in M_0 \text{ vs } H_1 : m \not\in M_1 \]

for two different models \( M_0, M_1 \) using the Integrated Regression Goodness of Fit as done in `intRegGOF`, but instead of the accumulation of the residual of a given model, in this case, the accumulation of the difference in the fits is considered:

\[
R_n^w(x) = n^{-1/2} \sum_{i=1}^n (\hat{y}_{0i} - \hat{y}_{1i}) I(x_i \leq x).
\]

The test statistics considered are \( \mathcal{K}_n \) and \( \mathcal{W}_n^2 \).

If `objH0` and `objH1` are `lm`, `glm` or `nls` fits for the models in classes \( M_0 \) and \( M_1 \) respectively, then `anovarIntReg(objH0, objH1)` computes test \( H_0 : m \in M_0 \text{ vs } H_1 : m \not\in M_1 \). When `anovarIntReg(objH0, objH1, ..., objHk)` is executed (notice that by default `INCREMENTAL=FALSE`) we obtain a table with the statistics \( K_n \) and \( W_n^2 \) and its associated \( p \)-values for each of the tests \( H_0 : m \in M_0 \text{ vs } H_i : m \not\in M_i \) being \( i = 1, \ldots, k \). On the other hand, if the parameter `INCREMENTAL` is set to TRUE, the command returns the results for the tests \( H_i : m \in M_i \text{ vs } H_{i+1} : m \not\in M_{i+1} \) being \( i = 1, \ldots, k-1 \).

Value

This function returns an object of class `anovarIntReg`, a matrix like structure whose rows refers to models and columns to statistics and its \( p \)-values. It also has an attribute `heading` to support printing the object.
Note

This method requires more testing, and careful study of the effect of factors (discrete random variables) when fitting the model.

Author(s)

Jorge Luis Ojeda Cabrera (<jojeda@unizar.es>).

See Also

lm, glm, nls, and intRegGOF.

Examples

```r
n <- 50
d <- data.frame(X1=runif(n),X2=runif(n))
d$Y <- 1 - 2*d$X1 - 5*d$X2 + rnorm(n, sd=.125)
a0 <- lm(Y~1,d)
a1 <- lm(Y-X1,d)
a2 <- lm(Y-X1+X2,d)
anovarIntReg(a0,a1,a2,B=50)
anovarIntReg(a0,a1,a2,B=50,INCREMENTAL=TRUE)
```

Description

Core functions for the computation of the Integrated Regression Goodness of Fit

Usage

```r
compIntRegProc(y, xord, weig = rep(1, length(y)))
compBootSamp(obj, datLT, B = 999, LINMOD = FALSE)
plotIntRegProc(y, x, weig = rep(1, length(y)), ADD = FALSE, ...)
getModelFrame(obj)
getResiduals(obj,type)
```

Arguments

- `y`: vector, values to add to compute the Integrated Regression.
- `xord`: list of list with the index of covariate points that are less than covariate data. This tells how to cumulate according to covariates.
- `weig`: vector of weights, specifically used to fit and compute test statistics when data is selection biased.
- `obj`: An object of class lm, glm or nls.
- `datLT`: structure as xord telling how to cumulate according to covariates.
intRegGOF

B

Bootstrap resampling size.

LINMOD

When TRUE and if obj is an object of class `lm` Linear Model matrix fitting equations are used.

x

vector with covarates to plot

ADD

If TRUE the plot is added to existing plot.

type

Type of residual.

... Further parameters to plot.

Details

...TODO: Each of them computes what in which way

Note

Surely they can better implemented.

Author(s)

Jorge Luis Ojeda Cabrera (<jojeda@unizar.es>).

Description

Integrated Regression Goodness of Fit to test if a given model is suitable to represent the regression function for a given data.

Usage

```r
intRegGOF(obj, covars = NULL, B = 499, LINMOD = FALSE)
## S3 method for class 'intRegGOF'
print(x,...)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
</table>
| obj      | An object of class `lm, glm` or `nls`.
| covars   | Names of continuous (numerical) variates used to compute Integrated Regression. They should be variables contained in the data frame used to compute the regression fit. |
| B        | Bootstrap resampling size. |
| LINMOD   | When TRUE and if obj is an object of class `lm` Linear Model matrix fitting equations are used. |
| x        | An object of class `intRegGOF`. |
| ...      | Further parameters for print command. |
The Integrated Regression Goodness of Fit technique is introduced in Stute(1997). The main idea is to study the process that results from the cumulation of the residuals up to a given value of the covariates. Once this process is built, different functional over it can be considered to measure the discrepancy between the true regression function and its estimation.

The tests that implements this function is

$$H_0 : m \in M$$ vs $$H_1 : m \notin M$$

being \(m\) the regression function, and \(M\) a given class of functions. The statistics considered are

$$K_n = \sup_{x \in \mathbb{R}^d} |R_n^w(x)|$$

$$W_n^2 = \int_{\mathbb{R}^d} R_n^w(z)^2 dF(z).$$

where \(R_n^w(z)\) is the cumulated residual process:

$$R_n^w(x) = n^{-1/2} \sum_{i=1}^n (y_i - \hat{y}_i) I(x_i \leq x).$$

As the stochastic behaviour of this cumulated residual process is quite complex, the implementation of the technique is based on resampling techniques. In particular the chosen implementation is based on Wild Bootstrap methods.

The method also handles selection biased data by means of compensation, by means of the weights used to fit the regression function when computing the cumulated residual process.

At the moment only 'response' type of residuals are considered, jointly with wild bootstrap resampling technique and the result for discrete responses might not be proper.

**Value**

This function returns an object of class \texttt{intRegGOF}, a list which contains following objects:

- \texttt{call} The call to the function
- \texttt{regObj} String with the \texttt{lm}, \texttt{glm} or \texttt{nls} object whose fit is checked
- \texttt{regModel} \texttt{lm}, \texttt{glm} or \texttt{nls} object call.
- \texttt{p.value} \textit{p}-values for \(K_n\) and \(W_n^2\) statistics.
- \texttt{datStat} value of \(K_n\) and \(W_n^2\) statistics.
- \texttt{covars} continuous (numerical) variates used to compute Integrated Regression.
- \texttt{intErr} cumulated residual process at the values of \texttt{covars} in data.
- \texttt{xLT} structure with the order of \texttt{covars} summation.
- \texttt{bootSamp} Bootstrap samples for \(K_n\) and \(W_n^2\).

**Note**

This method requires more testing, and careful study of the effect of factors (discrete random variables) when fitting the model.
Author(s)

Jorge Luis Ojeda Cabrera (<jojeda@unizar.es>).

References


See Also

lm, glm, nls and its methods summary, print, plot, etc...

Examples

n <- 50
d <- data.frame(X1=runif(n),X2=runif(n))
d$Y <- 1 + 2*d$X1 + rnorm(n,sd=.125)
plot( d )
intRegGOF(lm(Y~X1+X2,d),B=99)
intRegGOF(a <- lm(Y~X1-1,d),B=99)
intRegGOF(a,c("X1","X2"),B=99)
intRegGOF(a,-X2+X1,B=99)

Description

Methods to develop model validation and visualization of Integrated Regression Goodness of Fit technique.

Usage

plotAsIntRegGOF(obj, covar = 1, ADD = FALSE, ...)
pointsAsIntRegGOF(obj, covar = 1, ...)
linesAsIntRegGOF(obj, covar = 1, ...)
Arguments

- **obj**: An object of class `lm`, `glm` or `nls`.
- **covar**: Variable name, number or vector for which Int. Reg. is computed. If it is a number, it reference a covariate in the model frame, while if it is a name refer to data in data frame using in the fitting process.
- **ADD**: If TRUE the plot is added to existing plot.
- **...**: Further parameters to for plotobj command.

Details

Currently, the implementation computes the accumulated residual process against a single covariate (covar). When the value of covar is set to 0, the response is used as the variable whose residual are accumulated against.

Notice that if covar is a vector its lenght should be equal to the number of residuals.

Note

lm objects that does not have a data parameter set when the call is executed does not work presently when the covar parameter is different than 0.

Author(s)

Jorge Luis Ojeda Cabrera (<jojeda@unizar.es>).

See Also

`lm`, `glm`, `nls` its associated `plot` method and `intRegGOF`.

Examples

```r
n <- 50
d <- data.frame(X1=runif(n),X2=runif(n))
d$Y <- 1 + 2*d$X1 + rnorm(n,sd=.125)
par(ask=TRUE)
plot(d)
plotAsIntRegGOF(lm(Y~X1+X2,d),covar="X1")
plotAsIntRegGOF(a <- lm(Y~X1-1,d))
plotAsIntRegGOF(a,c("X1"))
plotAsIntRegGOF(a,0)
plotAsIntRegGOF(a,fitted(a))
par(ask=FALSE)
```
Utility Functions

Utility functions for Integrated Regression Goodness of Fit

Description

Functions that are basic or/and useful for the computation of the Integrated Regression Goodness of Fit

Usage

getLessThan(x, d)
mvCumSum(x, ord)
mvPartOrd(x1, x2)
getContVar(df, vars = NULL)
getModelCovars(obj)
getModelWeights(obj)
rWildBoot(n)

Arguments

x, d       matrix like structure.
x1, x2     vectors with the same length.
df         a data frame.
ord        list of list structure with the ordering to add data points according to a given covariates.
obj        An object of class \texttt{lm}, \texttt{glm} or \texttt{nls}.
vars       vector with variable names in observations data frame.
n          integer, sample size.

Details

...TODO: Each of them computes what in which way

Note

getLessThan can be certainly better implemented.

Author(s)

Jorge Luis Ojeda Cabrera (<jojeda@unizar.es>).
Index

* goodness of fit
  anovarIntReg, 1
  intRegGOF, 4
  plot, 6

* models
  anovarIntReg, 1
  intRegGOF, 4
  plot, 6

* nonlinear
  anovarIntReg, 1
  intRegGOF, 4
  plot, 6

* regression
  anovarIntReg, 1
  intRegGOF, 4
  plot, 6

anovarIntReg, 1

compBootSamp (Core Functions), 3
compIntRegProc (Core Functions), 3
Core Functions, 3

formula, 2

getContVar (Utility Functions), 8
getLessThan (Utility Functions), 8
getModelCovars (Utility Functions), 8
getModelFrame (Core Functions), 3
getModelWeights (Utility Functions), 8
getResiduals (Core Functions), 3
glm, 2–4, 6–8

intRegGOF, 2, 3, 4, 7

linesAsIntRegGOF (plot), 6
lm, 2–4, 6–8

mvCumSum (Utility Functions), 8
mvPartOrd (Utility Functions), 8

nls, 2–4, 6–8

plot, 6, 6, 7
plotAsIntRegGOF (plot), 6
plotIntRegProc (Core Functions), 3
pointsAsIntRegGOF (plot), 6
print, 6
print.anovarIntReg (anovarIntReg), 1
print.intRegGOF (intRegGOF), 4

rWildBoot (Utility Functions), 8

summary, 6

Utility Functions, 8